Minimization of Energy in Smart Phone Application Development Using Code Analysis

K. G. Srinivasa, Ch Brahm Prakash Government Engineering College, New Delhi, India
Srinidi Hiriyananaih, Ramaih Institute of Technology, Bangalore, India
G. M. Siddesh, Ramaih Institute of Technology, Bangalore, India

ABSTRACT

Mobile applications are popularly known as apps. Energy and battery-life are critical factors that enable the development and sustainability of apps on mobile devices. Application software developers need to consider the minimization of energy consumption along with the development and deployment of applications. Intelligent software engineering practices and tools are needed in order to assist developers in energy management of Android application development. This article proposes a rule-engine driven framework for estimating the energy consumption of an Android application by using program analysis of the source code. The basis of this framework is to provide the developer with a notion of which part of the application source code consumes considerable energy, and what alternatives could be used to replace it without changing its core functionality. It presents metrics at the overall, event and source code level, allowing application developers to optimize their applications early in the software development cycle.

KEYWORDS

Application Development, Code Analysis, Energy Efficiency, Smart Phone

1. INTRODUCTION

In the current world, people are surrounded with wide technologies. Mobile devices particularly, smartphones and tablets have become popular and used by millions of users across the globe for variety of purposes for daily lives. The growth market of smartphones is vibrant because of the strong application ecosystem. Smartphone applications provide numerous capabilities and functions to the users equipped with sensors, computing power, storage and communication via internet. However, the applications on the smartphone are restricted by the limited battery power. The recent advances in hardware components, operating systems, battery technology alleviates the limitation of the power usage in the smart phone. But, still the inefficiency of consumption of battery life from the applications cannot be prevented.

The energy consumption of software has significant impact on the overall energy consumption of its execution environment (Capra et al., 2012; Procaccianti et al., 2012). Research has been carried out on sophisticated power models for software development to estimate and predict the energy consumption through different parameters. In spite of this development, there is reusable information available for developers to create energy efficient software applications. (Procaccianti et al., 2016). Despite spreading awareness among developers about the importance of limiting the consumption of

DOI: 10.4018/IJOSSP.2018070104

Copyright © 2018, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
energy in an app, they lack guidance on how energy is typically consumed by mobile applications, and follow unknowingly follow practices that are detrimental in this regard. Thus, there is a need to aid the application development to engineer energy efficient applications for developers.

Energy usage patterns help developers to gain insight and assist the development using existing tools. Examples of such tools include cycle-accurate simulators, power monitors, program analyses and statistical measurement techniques. These tools lack the direct guidance of energy consumption of applications and assist development. They do not provide an understanding of energy pattern usage and how the code can be tweaked to reduce the energy (Hao, Li, Halford & Govindan, 2014). This paper proposes a framework that attempts to bring this information to the developer driven by metrics of energy consumption of an Android app.

The proposed framework provides the developer with enough information to plan app development while adhering to reasonable energy constraints. It does not require external hardware to be connected to the smartphone, unlike most existing techniques that measure energy consumption of an app. Analysis of the source code of the app is carried out statically and do not require the developer to produce execution test cases to simulate the apps performance. Thus, at any stage of the Software Development Life Cycle, the developer is empowered to view the apps current energy score. The framework utilizes a flexible rule engine, which is built to acquire energy related information from various source files that are part of the application directory.

2. RELATED WORK

Energy acts as a critical resource for smart phones. The quantitative and objective information of consumption of energy is needed for the development of applications for smart phones. In this section, a brief review of the existing research on the energy awareness and consumption for smart phones are described.

Before the advent of smart phones, computer applications and software were predominant. Energy consumption of this software is critical and can be evaluated in many aspects. Software design patterns can be used to determine the energy consumption of the applications using software design patterns (Sahin, C., et.al 2015) It analyzes the impact of energy from low-level profiles to high-level architecture and its objects and modules. An empirical study on different code styles and its impact is depicted with the help of class diagrams, sequence diagrams and the amount of messages passed among them. This framework is built for traditional software systems and does not apply to smartphones, but provides a basis that can be applied on smartphones.

Energy consumption in smart phones depends on various aspects and components including CPU, memory, graphics, storage, audio and networking interfaces, under varying application loads by hardwiring individual power meters to different phone components (Carroll & Heiser, 2010). The energy consumption was analysed in detail at a hardware component level with wide range of realistic usage scenarios and validated the results with multiple mobile devices. (Dong & Zhong, 2011; Zhang et al., 2010) has proposed techniques that provide energy consumption estimates to users by observing important operating system parameters at run time. These estimates are provided at a higher level of granularity – either the method or component level which are too generalized to provide developers with enough information to perform changes. The proposed system in this paper uses this as a basis but with more extension to the code-level analysis rather than the hardware component level.

Energy profiling is one of the other methods for used for determining energy consumption in the smart phones. A tool, called eProf, was created for fine-grained energy profiling for smartphone applications (Pathak et al., 2010) It traces system calls made by applications and uses power state models for various components to infer energy used. It can incorporate tail-energy use, such as when a component remains in a higher energy state after the application is done using it. eProf delineates the domain of no-sleep bugs in smartphone apps, characterizes them completely and presents the first compile-time technique for detecting them, based on the reaching definitions dataflow algorithm.
Are Developers Fixing Their Own Bugs?: Tracing Bug-Fixing and Bug-Seeding Committers

Open Source as a Strategic Asset: Evidence from the Financial Industry